TXTree

Introduction

TXTree is a vector-based analysis tool designed to transform a MEDLINE file (a PubMed search result) into an HTML-TM (Hypertext Markup Language for Text Mining), a portable HTML platform for exploring texts.

Installation

TXTree is available as a command-line tool, with installers for both Linux and Windows, and can be downloaded from SourceForge.

Exploring the HTML-TM Interface

The HTML-TM interface is designed to be intuitive and user-friendly, enabling researchers and analysts to explore large text datasets easily. It provides two main pages for analysis: **WORDS.html** and **TEXTS.html**, each offering unique insights into the dataset.

The **WORDS.html** page focuses on vocabulary analysis, displaying details such as word occurrences, related words, and semantic connections through a hierarchical Word Tree. It also includes a Year Plot to visualize word usage trends over time. The **TEXTS.html** page, on the other hand, allows users to explore individual documents, showing titles, publication years, and links to related documents based on cosine similarity. Both pages direct users to a **Related Documents Page**, which provides detailed information about documents most similar to a target word or document, including titles, abstracts, and PubMed IDs.

To enhance usability, the HTML-TM interface includes a search utility. Users can perform queries using logical operators (AND/OR), target specific columns (e.g., word, title, or year), and utilize regular expressions (Regex) for pattern matching.

Obtaining a MEDLINE File

1. Downloading from the PubMed Website

To download MEDLINE records from PubMed, after your search, click “Save”, change the default option from “Selection” to “All results”, and select “PubMed” as the format. Note the 10,000-record limit.

2. Using Entrez Direct (EDirect) to Bypass the Limitation

To bypass the **10,000 results** limitation, Entrez Direct (EDirect) can be used. EDirect is a suite of Unix-based command-line tools provided by NCBI for accessing and retrieving data from their databases. While EDirect is natively designed for Linux, it can be used on Windows by leveraging the **Windows Subsystem for Linux (WSL)**. WSL allows users to run a Linux environment directly on Windows, enabling the seamless execution of EDirect commands.

Installation instructions for EDirect are available in the EDirect Documentation. Below is an example of a command executed in the terminal to retrieve MEDLINE-formatted results for a specific query:

```
esearch -db pubmed -query '"nitrogen fixation"' | efetch -format medline
        > nitrogen_fixation.medline
```

TXTree Usage

The basic syntax for running TXTree is:

```
txtree [OPTIONS] input_path
```

Help Argument

- `-h, --help`: Show this help message and exit.

Required Argument

- `input_path`: Path to the MEDLINE dataset file or a preprocessed directory.
  Note: The preprocessed directory functionality allows users to reuse previously generated files (e.g., tokenized data, TF-IDF scores, or embeddings) to continue execution from where files are saved. This avoids redundant processing and saves time when resuming or extending analyses.

Output Argument

- `--output_dir OUTPUT_DIR`: Specifies the directory to store output files.
  Default: `txtree_result`.

HTML-TM Interface Options

- `--html_tm_title HTML_TM_TITLE`: Sets the title for the HTML-TM. Default: `"HTML-TM"`.

- `--html_tm_theme {dark,light}`: Defines the theme for the HTML-TM interface.
  Default: `dark`.

Filtering Option

- `--tf_idf_threshold TF_IDF_THRESHOLD`: Filters words based on their TF-IDF scores. Words below this threshold are removed. Default: `0.1`.

Dendrogrammatic Ordination Options

- `--word_ord_max_clus WORD_ORD_MAX_CLUS`: Maximum number of clusters for word ordination. Default: None.
- `--doc_ord_max_clus DOC_ORD_MAX_CLUS`: Maximum number of clusters for document ordination. Default: None.

Optional Output

- `--save_emb`: Save embedding vectors in an HDF5 file in the output directory. Default: `False`.

File Recreation Options

- `--force_xml`: Force recreate XML file. Default: `False`.
- `--force_word_processor`: Force recreate Word Processor file. Default: `False`.
- `--force_temporal_correlation`: Force recreate Temporal Correlation file. Default: `False`.
- `--force_html_tm`: Force recreate HTML-TM files. Default: `False`.

Directory Management

- `--del_exist_dir`: Deletes the existing output directory if it exists. Default: `False`.

Suppression Options

- `--suppress_temporal_correlation`: Disables the Temporal Correlation process. Default: `False`.
- `--suppress_html_tm`: Disables the creation of HTML-TM files. Default: `False`.
- `--suppress_html_tm_words`: Disables the creation of the WORDS.html file. Default: `False`.
- `--suppress_html_tm_texts`: Disables the creation of the TEXTS.html file. Default: `False`.

Special Mode

- `--only_emb`: Generate only word embeddings, skipping Temporal Correlation and HTML-TM creation. This automatically sets `save_emb` to `True`, and both `suppress_temporal_correlation` and `suppress_html_tm` to `True`. Default: `False`.

Memory Management

- `--ignore_memory_check`: Skip memory validation before processing. Use with caution, as this may cause crashes due to insufficient RAM. Default: `False`.

Parallelization Options

- `--n_jobs N_JOBS`: Number of parallel jobs for applicable tasks. Default: `1` (no parallelization).
- `--chunk_size CHUNK_SIZE`: Chunk size for parallel execution. Larger values may reduce communication overhead but increase memory usage. Default: `1000`.

Testing Mode

- `--test_html_tm`: Activates test mode for HTML-TM generation without complete processing. Default: `False`.

Verbosity

- `--quiet`: Disables verbose output. Default: `True`.

TXTree Output

HTML-TM Visualization (Ready-to-Use Platform)

Self-contained folder for end-users. The only output needed for distribution.

- `html_tm/`
    - `WORDS.html` - Interactive term explorer:
        - Displays top related terms + hierarchical clusters;
        - Links to 20 most relevant documents per term.
    - `TEXTS.html` - Document search tool:
        - Find similar articles using shared terms;
        - Consistent numbering with WORDS.html.
    - Supporting files:
        - Precomputed data (JSON/CSV);
        - JavaScript/CSS for the interface.

Core Processing Files

Required for reprocessing but not needed by end-users. Recommended to delete before distribution, as these files are large and unnecessary for the final visualization.

- `dataset.xml`
    - Structured XML version of the input MEDLINE data.
- `word_processor.pkl`
    - Processed words/documents, TF-IDF scores, and embeddings.
- `temporal_correlation.pkl`
    - Temporal trends in word usage.

Metadata & Documentation

Supplementary files for reference.

- `words.txt` - Final list of filtered/ordered terms.
- `doc_ids.txt` - Sorted document IDs.
- `parameters.txt` - Settings used.

Optional Advanced Output

Generated only if requested (save_emb=True).

- `embeddings.h5`
    - Raw word/document embeddings + PCA coefficients.

Example Commands

Basic Execution

```
txtree dataset.medline
```

This command processes `dataset.medline` using default settings, generating word embeddings, Temporal Correlation data, and an HTML-TM visualization.

Specifying an Output Directory

```
txtree --output_dir results dataset.medline
```

Stores the output files in the `results` directory instead of the default `txtree_result`.

Defining a Custom Title for HTML-TM

```
txtree --html_tm_title "My Analysis" dataset.medline
```

Sets the title displayed at the top of the HTML-TM interface to “My Analysis”.

Generating Only Embeddings

```
txtree --only_emb dataset.medline
```

Runs TXTree to extract embedding vectors without generating Temporal Correlation or HTML-TM files.

Forcing Regeneration

```
txtree --force_html_tm dataset.medline
```

Recreates the HTML-TM visualization even if it already exists in the output directory.

Running with Parallel Processing

```
txtree --n_jobs 4 --chunk_size 2000 dataset.medline
```

Processes the dataset using four parallel jobs with a chunk size of 2000 for improved performance on multi-core systems.

Skipping Memory Check

```
txtree --ignore_memory_check large_dataset.medline
```

Processes a large dataset while skipping the memory check (use only if sufficient RAM is available).